

1 **A METHOD AND A SYSTEM FOR CERTIFICATE REVOCATION LIST**
2 **CONSOLIDATION AND ACCESS**

3 **FIELD OF THE INVENTION**

4 This invention relates to digital certificates, and, more
5 particularly, to certificate revocation list consolidation and
6 access.

7 **BACKGROUND OF THE INVENTION**

8 Systems for accomplishing business transactions
9 electronically are becoming increasingly widespread, partly
10 because of the advent of global computer networks such as the
11 Internet, and partly because of the evolution and maturity of
12 public key cryptography, which enhances the security of such
13 commerce. The application of public key cryptography to
14 electronic commerce has been heretofore envisioned in
15 documents such as Recommendation X.509 of the International
16 Telecommunications Union (ITU, formerly CCITT).

17 To use secure electronic commerce according to the
18 conventional methods, each user has a pair of related keys,
19 namely a private key and a public key. A public key is simply
20 a value (generally a number), and has no intrinsic association
21 with anyone, including the person whose message it is to
22 authenticate. Widespread, commercial use of digital signatures
23 requires reliable information associating public keys with
24 identified persons. Messages of those identified persons can
25 then be authenticated using the keys.

1 Digital signature certificates meet this need. These
2 certificates are generally issued by trusted third parties
3 known as certification authorities (CAs) and they certify (1)
4 that the issuing certification authority has identified the
5 subject of the certificate (often according to specifications
6 delineated in a certification practice statement), and (2)
7 that a specified public key (in the certificate) corresponds
8 to the a private key held by the subject of the certificate. A
9 structure for public-key certificates is included in the X.509
10 standard cited earlier. The content of a certificate is often
11 specified in a statute or regulation. A typical X.509
12 certificate has the format:

X.509 Certificate

Version Number
Certificate Serial Number
Algorithm Identifier
Certificate Issuer
Validity Period
Subscriber
Subscriber's public key information
Signature of Issuer

13

1 Generally, a valid term is specified in the certificate.
2 The certificates become invalid for its expiration. In
3 addition, the invalid certificates may include any certificate
4 which:

5 (1) has been revoked (i.e., have been declared
6 permanently invalid by the certification authority which
7 issued the certificate); and

8 (2) is suspended at the time of reliance (i.e. has been
9 declared temporarily invalid by the certification authority
10 which issued the certificate).

11
12 Suspending and/or revoking certificates are an important
13 means of minimizing the consequences of errors by the
14 certification authority or subscriber. Depending on applicable
15 legal rules, a certification authority may avert further loss
16 due to inaccuracy in the certificate by revoking it. A
17 subscriber can revoke a certificate to prevent reliance on
18 forged digital signatures created using a compromised, e.g.,
19 lost or stolen, private key. Certificates which become invalid
20 by revocation are generally listed in a certificate revocation
21 list (CRL), according to ITU X.509. Suspension, or temporary
22 invalidation, was not contemplated in ITU X.509, and may or
23 may not be included in the CRL. Certificates which become
24 invalid by virtue of their age need not be listed in a CRL
25 because each certificate contains its own expiration date.

26 As a practical matter, the conventional CRL-based system
27 works as follows. Before a subscriber can create a verifiable
28 digital signature, the signer must arrange for a certification
29 authority to issue a certificate identifying the subscriber

1 with the subscriber's public key. The subscriber receives back
2 and accepts the issued certificate, and then creates digital
3 signatures and attaches a copy of the certificate to each of
4 them. When the other party to a transaction receives such a
5 digital signature, the other party must check with the
6 certification authority, generally via its on-line database,
7 to determine whether the certificate is currently valid. If
8 so, and if the digital signature can be verified by the public
9 key in the certificate, the party is usually in a strong
10 position to rely on the digital signature.

11
12 Modern e-business typically work or will work on the open
13 Internet and need to access CRLs from multiple CAs as users
14 may use any existing CAs of their choice. But different CAs
15 may use different CRL distribution mechanisms, some of which
16 are very complicated. This demands that the application
17 developers have rich knowledge of these CRL distribution
18 mechanisms. In addition, some CAs may change their CRL
19 distribution mechanisms from time to time, which may impose
20 significant change on the way in which applications can access
21 their CRL.

22 Further, CRL online distribution, such as via a directory
23 server, is adopted by some CAs. An application can down load
24 these CRLs when needed. But real time access to CRL may be
25 very expensive and not necessary for most application. In
26 addition, the CRLs downloaded and parsed by one application
27 can't be shared by others, which is also a waste of system
28 resources.

1 SUMMARY OF THE INVENTION

2 In order to solve the above problems, this invention
3 provides methods and systems for certificate revocation list
4 consolidation and access. This invention uses different CRL
5 retrieval agents for different CRL distribution methods to
6 consolidate CRLs from multiple CAs into a central CRL
7 database, which can be replicated to other machines via
8 network. An application can access the nearest CRL database to
9 determine whether a digital certificate has been revoked via a
10 set of unified APIs without bothering the details of CRL
11 distribution, retrieval and representation. In addition, since
12 the CRL database can be shared by all the applications, the
13 system resources is used efficiently.

14 According to an aspect of the invention, a system for
15 certificate revocation list consolidation and access is
16 provided. In an example embodiment the system comprises:

17 a plurality of certificate authorities (CAs), in which
18 each CA maintains and distributes the digital certificates
19 revoked by itself in the form of CRL, and different CAs may
20 use different CRL distribution mechanisms;

21 a plurality of CRL databases, for storing the
22 consolidated CRLs from multiple CRL retrieval agents or the
23 replications of CRLs; and

24 CRL access user interface, for providing a uniform set of
25 APIs for user's accessing the CRLs CRL databases.

1 According to another aspect of the invention, a method
2 for certificate revocation list (CRL) consolidation and access
3 is provided, wherein a plurality of certificate authorities
4 (CAs) maintain and distribute the digital certificates revoked
5 by themselves in the form of CRLs, and different CAs may use
6 different CRL distribution mechanisms, said method comprising
7 the steps of:

8 creating a plurality of CRL retrieval agents based on the
9 CRL distribution mechanisms of CAs, for consolidating the CRLs
10 from multiple CAs;

11 storing the consolidated CRLs from multiple CRL retrieval
12 agents or the replications of CRLs into a plurality of CRL
13 databases; and

14 accessing the CRLs from the CRL databases by a uniform
15 set of APIs.

16 **BRIEF DESCRIPTION OF THE DRAWINGS**

17 Additional objects and advantages of this invention will
18 be apparent from the following detailed description of
19 preferred embodiments thereof which proceeds with reference to
20 the accompanying drawings, in which:

21 Figure 1 is a block diagram illustrating the system for
22 certificate revocation list consolidation and access according
23 to a preferred embodiment of this invention;

24 Figure 2 illustrates the operation of LDAP/CRL retriever
25 agent according to a preferred embodiment of this invention;

1 Figure 3 illustrates the operation of HTTP/CRL retriever
2 agent according to a preferred embodiment of this invention;

3 Figure 4 illustrates the operation of RFC 1424/CRL
4 retriever agent according to a preferred embodiment of this
5 invention;

6 Figure 5 illustrates the operation of Http receiver agent
7 according to a preferred embodiment of this invention;

8 Figure 6 illustrates the architecture of the CRL database
9 replication;

10 Figure 7 is a block diagram illustrating the system for
11 certificate revocation list consolidation and access according
12 to another preferred embodiment of this invention; and

13 Figure 8 is a flow diagram illustrating the system for
14 certificate revocation list consolidation and access according
15 to another preferred embodiment of this invention.

16 **DESCRIPTION OF THE INVENTION**

17 Before describing the preferred embodiment of this
18 invention, first we describe the general problem of the
19 certificate revocation list consolidation, such as, the format
20 of CRLs, the distribution mechanism of CRLs, etc.

21 As described in the above, Digital Certificate is a
22 digital document attesting that a particular public key

1 belongs to a particular entity. The most widely accepted
2 format for certificates is defined by the ITU-T X.509
3 international standard. A Digital Certificate:
4 • is digitally signed by some trusted entity called
5 Certificate Authority (CA);
6 • is valid only within a certain designated period of
7 time;
8 • can be verified by anyone having access to its signing
9 CA's public key; and
10 • is labeled with a unique serial number for identifying
11 it within its issuing CA.

12 However, a digital certificate may have been revoked for
13 some reason within the validity period of the digital
14 certificate. Hence, each CA has the obligation to maintain a
15 Certificate Revocation List (CRL), make it publicly available
16 and refresh it in certain time intervals. Each revoked digital
17 certificate included in a CRL can be identified by its serial
18 number.

19 X.509 CRL is defined as an ASN.1 SEQUENCE structure as
20 below,

21 CertificateList ::= SEQUENCE{
22 tbsCertList TBSCertList,
23 signatureAlgorithm AlgorithmIdentifier,
24 signature BIT STRING
25 }

26 where signatureAlgorithm identifies the algorithm used by
27 the signing CA for computing the digital signature from the
28 ASN.1 DER encoded TBSCertList structure, which itself is also
29 expressed in an ASN.1 SEQUENCE structure as specified below.


```

1      TBSCertList ::= SEQUENCE{
2          version          Version OPTIONAL,
3          signature        AlgorithmIdentifier,
4          issuer           Name,
5          thisUpdate       Time,
6          nextUpdate       Time OPTIONAL,
7          revokedCertificates SEQUENCE OF SEQUENCE{
8              userCertificate    CertificateSerialNumber,
9              revocationDate     Date,
10             crlEntryExtension  Extensions OPTIONAL
11         }OPTIONAL
12         crlExtensions[0]     EXPLICIT Extensions OPTIONAL
13     }

```

14 The TBSCertList specifies the distinguished name of the
15 issuer, the issuing date of the CRL, the date when the next
16 CRL will be issued, and, optionally, lists of revoked digital
17 certificates (identified by their serial numbers) and CRL
18 extensions.

19 Two points should be noted here. First of all, the list
20 of revoked digital certificates is optional because a
21 particular CA might not have revoked any certificates it has
22 issued so far when its CRL publication is due. Secondly,
23 although a CA might specify in its CRL the next scheduled
24 publication date of CRL, this does not prevent the CA from
25 publishing its CRL on a more frequent basis, e.g., in case of
26 emergency. Nevertheless, this optional nextUpdate convention
27 enables users to get a sense when a given CRL is "out of
28 date".

29 Generally, there are two models for a CA to distribute
30 its CRL. In the "pull" model, verifiers (users who want to
31 verify the status of certain digital certificate(s)) can
32 download the CRL from the CA when needed. In the "push"

1 model, the CA sends the CRL to the registered verifiers at
2 regular intervals.

3 The authentication framework defined by X.509 is
4 originally designed to operate in the X.500 directory
5 environment. X.500 makes provision for the storage of CRL's as
6 directory attributes associated with CA entries. X.500 also
7 defines the DAP (Directory Access Protocol) used by clients
8 to access the directory. However, DAP is significantly more
9 complicated than the more prevalent TCP/IP stack
10 implementations and requires more coding and computing
11 horsepower to run. The size and complexity of DAP make it
12 difficult to run on smaller machines such as the PC's. LDAP
13 (Lightweight Directory Access Protocol) is designed to remove
14 some of the burden of X.500 access from directory client,
15 making the directory available to a wider variety of machines
16 and applications. LDAP runs directly over TCP/IP (or other
17 reliable transport) implementations bundled with most modern
18 machines today. Verifiers can use the "pull" model to
19 retrieve CRL's from LDAP servers.

20 However, X.500 directory service is not expected to be
21 ubiquitous on the Internet in the near future. Some other
22 affordable CRL distribution and access methods using existing
23 Internet infrastructure are developed to address the needs
24 today. One such method is defined in Privacy-Enhanced Mail.
25 According to Privacy-Enhanced Mail, Internet Policy
26 Registration Authority (IPRA) should coordinate with Policy
27 Certification Authorities (PCA's) to provide a robust
28 database facility which contains CRL's issued by the IPRA, by
29 PCA's , and by all other CA's . Access to this database is
30 provided through mailbox facilities maintained by each PCA.
31 The verifiers can retrieve CRL's from their specified one or
32 more e-mail addresses using the mechanisms defined in RFC1424.

1 The CRL retrieval methods works either in "pull model or in
2 "push model, depending on whether or not the PCA's concerned
3 support unsolicited distribution of CRL's .

4 In the above, the general problem of the certificate
5 revocation list consolidation, such as, the format of CRLs and
6 the distribution mechanism of CRL have been discussed. Now,
7 the system and the method for certificate revocation list
8 consolidation and access will be disclosed with reference to
9 the accompanying drawings.

10 In the system of this invention, the central CRL database
11 is created for the storage of consolidated CRLs. For different
12 CRL distribution method, different CRL retrieval agents are
13 used to retrieve periodically CRL's from different CAs and
14 consolidate them into the central CRL database. A set of
15 unified API's is provided for online applications to access
16 CRL information.

17 Figure 1 shows our Lotus Domino based framework of CRL
18 accessing middleware. Three CRL retrieval methods are
19 supported at this time, while more CRL retrieval methods can
20 be integrated with relative ease. The revoked certificates are
21 extracted from the retrieved CRL's and are stored in a Domino
22 database called Central CRL Database, which is to be
23 replicated to other databases called CRL Database Replica on
24 other connected Domino servers. A set of Java based CRL access
25 API's are also provided for e-Commerce applications to
26 actually take advantage of the consolidated CRL's at the
27 nearest Domino server, without bothering the details of how
28 the CRL's are issued by various CA's .

29 Although different CA's may use different CRL retrieval
30 methods, all such methods must include processes to download
31 CRL's from specified locations, to verify downloaded CRL so as
32 to ensure it is indeed issued by the intended CA, and to save

1 the downloaded CRL to a central CRL database. As one example
2 in our case, a specific CA puts its CRL in a LDAP directory, a
3 Domino Agent residing in the central CRL database is scheduled
4 to run periodically to retrieve CRL from the designated LDAP
5 server via LDAP protocol and update the central database
6 accordingly. Any change in the central CRL database will be
7 replicated to other CRL database replica. This not only makes
8 the CRL database management easier but also enables an
9 e-Commerce application to have easier, faster, and less costly
10 access to CRL database. When an e-Commerce application wants
11 to determine whether a certificate has been revoked, they only
12 need to make a CRL access API call to the nearest CRL Database
13 Replica. The CRL access API then calls NOI (Notes Object
14 Interface) to access the Domino CRL database, ascertain
15 whether the certificate is listed in the CRL and return the
16 result to the e-Commerce application.

17 **1. The Central CRL Database**

18 The CRL Database bases its design on a CRL Database
19 template, which comprise a number of forms, views, and agents
20 described below.

21 • Domino Forms for CRL Database

22 The CRL Database contains documents created from three
23 forms: a Trusted Certificate Authority form, a Revoked
24 Certificate form and a Memo form. The Trusted Certificate
25 Authority form mainly includes the fields presented in table
26 1.

27 The DistinguishedName field represents the distinguished
28 name of a CA conforming to RFC 1779, which defines a
29 user-oriented string representation of distinguished name. The
30 Certificate field holds the CA's X.509 v3 certificate that is
31 in Base64 encoded DER format. The certificate is used to

Field Name	Stored Data
DistinguishedName	Distinguished Name of a CA conforming to RFC 1779 ^[6]
Certificate	X.509 Certificate of a CA
ThisUpdate	the last time when a CA updated its CRL
NextUpdate	the next time when a CA is to update its CRL
CRLNumber	the current CRL sequence number of a CA
LDAPURL	a LDAP URL conforming to RFC 2255 ^[7]
PCAMailbox	e-mail address of a PCA for RFC 1424 CRL service

Table 1: Trusted Certificate Authority Form Definition

verify certificates and CRL's signed by this CA. To avoid input errors, the source of this field can be from the Clipboard or from a local certificate file. The ThisUpdate, NextUpdate and CRLNumber fields get their values from the latest retrieved CRL. The NextUpdate and CRLNumber fields may be empty. The CRLNumber is a CRL extension conveying a monotonically increasing sequence number for each CRL issued by a given CA. This extension allows users to easily determine whether a particular CRL supersedes another CRL. The LDAPURL field contains a LDAP URL conforming to RFC 2255 ^[7], which is used by the LDAPRetriever Agent to retrieve CRL from the specified LDAP server. The PCAMailbox field contains the

1 e-mail address of the CA's PCA for RFC 1424 CRL service. If
2 the PCA supports unsolicited CRL distribution, this field may
3 be empty.

4 Each revoked certificate extracted from the retrieved CRL
5 is stored in a document created from the Revoked Certificate
6 form that mainly includes the following fields.

Field	Stored Data
Distinguished Name	Distinguished Name for a CA conforming to RFC 1779
Serial Number	serial number of a revoked certificate
Revoke Date	date when a certificate is revoked
Revocation Reason	reason for revoking a certificate

7
8 Table 2: Revoked Certificates Form Definition

9 All fields, except for the RevocationReason, in the
10 RevokedCertificate fields are mandatory. The DistinguishedName
11 field and the SerialNumber field unequivocally identify a
12 revoked certificate. The RevocationReason field describes a
13 CRL entry extension reasonCode, which is used to identify the
14 reason for revoking a certificate.

15 The Memo form is for the RFC1424 PEM messages. The form
16 mainly includes the following fields.

Field	Stored Data
From	e-mail address of PEM message sender
To	e-mail address of PEM message receiver
Date	PEM Message sent date
Subject	PEM Message subject
Body	PEM message body

• **Domino Views for CRL Database**

Three views, a Trusted Certificate Authorities view and a Revoked Certificates \ By Issuer view and a Revoked Certificates \ By Serial Number view, are created in this Domino database in order to speed up searching for a particular CA and/or revoked certificate.

The Trusted Certificates Authorities view has columns titled Distinguished Name, Current CRL Update Time, Next CRL Update Time and Current CRL Number, which get their values from the corresponding fields of every Trusted Certificate Authority document. In these columns, Distinguished Name is the auto-sorted column.

The Revoked Certificates \ By Issuer view has columns titled Distinguished Name, Serial Number, Revoked Date and Revocation Reason, which get their values from the corresponding fields of every Revoked Certificate document. In these columns, Distinguished Name is the primary sorting column and Serial Number is the secondary sorting column. In addition, Distinguished Name is also the Categorized column.

The Revoked Certificates \ By Serial Number view has the same titled columns with the Revoked Certificate \ By Issuer view, but in a different order, Serial Number, Distinguished

1 Name, Revoked Date and Revocation Reason. In these columns,
2 Serial Number is the primary sorting column and Distinguished
3 Name is the secondary sorting column.

4 • The Domino Agents

5 The CRL Database also has the following Java Domino agents
6 named LDAP Retriever, HTTP Retriever, RFC1424 Requester,
7 RFC1424 Receiver and HttpReceiver, respectively. The LDAP
8 Retriever Agent, HTTP Retriever agent and RFC1424 Requester
9 Agent are background agents, the former periodically retrieves
10 CRL's of trusted CA's from LDAP servers or X.500-LDAP gateways
11 and stores them in the CRL Database, the latter sends RFC1424
12 CRL retrieval request messages to PCA mailboxes in certain
13 time interval. The RFC1424Receiver Agent is activated when new
14 mail has arrived, then it retrieves CRL's from the received
15 mail and stores them in the CRL Database. The HttpReceiver
16 Agent is triggered by a HTTP request. It verifies the
17 authorization of the requester. If successful, the agent
18 stores the transmitted CRL in the CRL Database, so the HTTP
19 task must be running in the hosting Domino server. This agent
20 provides a convenient way to incorporate other external CRL
21 retrieval methods, which only need to transmit the CRL's they
22 have received in a HTTP POST message as below:

23 POST /X509CRL.nsf/HttpReceiver?OpenAgent HTTP/1.0
24 Content-length: <content length>
25 Content-type: application/pkix-crl
26 Content-transfer-encoding: base64

27 <base64 encoded CRL>

28 However, the LDAPRetriever Agent is an unrestricted agent
29 as it will perform network I/O operations, so you might have
30 to modify the server record in the public Name and Address
31 book to enable the agent to run on the server.

1 **2. LDAP retriever agent**

2 As shown in Figure 2, the LDAP Retriever Agent connects
3 to LDAP servers to retrieve CRL and update the CRL Database
4 accordingly.

5 Currently, there are two Java interfaces to LDAP, JDAP
6 and JNDI. JDAP is an LDAP class library defined in an IETF
7 draft ^[8]. JDAP is supported in Netscape Directory SDK for
8 Java. Java Naming and Directory Interface (JNDI) is part of
9 the Java Enterprise API set, supported by many vendors
10 including IBM, HP, Novell etc. However, our discussion below
11 is neutral to either API.

12 LDAP servers may require a bind operation to authenticate
13 client identity before any other LDAP operations. In the
14 normal case, the CRL attribute of a CA entry is publicly
15 available, hence anonymous bind operation is good enough. LDAP
16 V2 clients have to send a Bind Request in the first Protocol
17 Data Unit(PDU) of the connection, while LDAP V3 clients do
18 not need to perform bind operation, since LDAP V3 servers
19 automatically treat operations without prior binding as
20 anonymous operations ^[3]. In order to keep compatibility with
21 LDAP V2 server, we always request an anonymous bind operation
22 prior to performing any other LDAP operations.

23 After the bind operation, the LDAP Retriever Agent uses the
24 specified LDAP URL to get the CA's latest CRL from the LDAP
25 server. And then, the LDAP Retriever Agent updates the CRL
26 Database with the retrieved CRL.

1 **3. HTTP retriever agent**

2 The operation of HTTP retriever agent is similar to the
3 operation of LDAP retriever agent. As seen from Figure 3, HTTP
4 retriever agent retrieves periodically CRL's from CAs.

5 **4. RFC1424 retriever agent**

6 As we discussed in the above 3, RFC1424 CRL retrieval
7 service is provided through mailboxes maintained by each CA's
8 PCA. If you want to get a CA's latest CRL, you need to
9 register with the PCA or send a CRL-retrieval request to the
10 PCA's mailbox. The PCA will send you a CRL-retrieval reply
11 message containing the requested CRL. Both CRL-retrieval
12 request message and CRL-retrieval reply message are a type of
13 Privacy-Enhanced Message(PEM). So you must have a mailbox and
14 a PEM user agent to send CRL-retrieval request messages and to
15 receive CRL-retrieval reply messages.

16 Domino Mail-In database record within the public Name and
17 Address book provides a means of receiving e-mails directly
18 into a Notes application, which is generally referred to as
19 mail enabled application. The Central CRL Database is such a
20 mail enabled application. As discussed in section 2.2.3, two
21 agents residing in the CRL database, RFC1424 Receiver and
22 RFC1424 Requester, fulfill the task of accessing RFC1424 CRL
23 service. The figure 3 depicts this.

24 If the PCA's support unsolicited CRL distribution, i.e.,
25 when the latest CRL's are available, the PCA's automatically
26 send the CRL-retrieval reply messages to your mailbox, the
27 scheduling of the RFC1424 Requester Agent may be disabled.

28 The RFC1424 Requestor Agent listens for incoming CRL
29 retrieval reply messages, verifies the retrieved CRL's and
30 stores them in the CRL Database.

1 Since PCA usually uses standard Internet mail address, the
2 hosting Domino Server for CRL Database must be able to
3 exchange e-mail messages with the Internet e-mail servers.

4 **5. Http receiver agent**

5 As shown in from Figure 5, Http receiver agent is
6 triggered by HTTP request. The Http receiver agent verifies
7 the request, if successful, stores CRL sent by them in the CRL
8 Database.

9 **6. CRL Database Replication**

10 To distribute the CRL Database over the entire Notes
11 network, we make use of the Notes database replication
12 functionality to replicate the CRL Database to other Domino
13 server. e-Commerce applications can have easier, faster and
14 less costly access to CRL from the nearest(even local, in
15 some cases) CRL Database Replica.

16 As shown in Figure 6, the Hub-and-Spoke replication
17 architecture is set up to fulfill replication task, as shown
18 in Figure 4. The hub server is responsible for,

- 19 • retrieving the latest CRL's from all trusted CA's
- 20 • updating the local CRL Database
- 21 • propagating the update to the spoke servers

22 Although you can specify what replication types should be
23 specified: Pull-Push, Pull-Pull, Push-Only and Pull-Only, it
24 is apparent that Push-Only or Pull-Only type is appropriate
25 for the situation, because there is no change needed to
26 propagate from the spoke servers to the hub server. Specify
27 Push-Only or Pull-Only only affects which server initiates the
28 replication work: either the hub server pushes or the spoke

1 servers pull. Only the replication connection records and the
2 database ACL need to be modified accordingly.

3 For each spoke server, a replication connection record must
4 be created in the public Name and Address book. In all of
5 these records, if the Replication Type field in Routing and
6 Replication section is set to Push-Only, the Source server
7 field and Destination server field in Basics section should be
8 specified as hub server and spoke server respectively . If the
9 Replication Type field is set to Pull-Only, the source server
10 must be specified as spoke server, the destination server must
11 be specified as hub server.

12 For Push-Only model, the CRL Databases in the spoke servers
13 must assign at least the Designer right to the hub server.
14 However, for Pull-Only model, the CRL Database in the hub
15 server only need to assign Reader right to the spoke servers.
16 So the Pull-Only replication model can be recommended.

17 **7. CRL Access API in Java**

18 Our system not only retrieves and consolidates CRL's from
19 multiple CA's , it also provides a set of Java CRL access API
20 for e-Commerce applications. The API's are represented as a
21 Java class CRLAccessAgent. The constructor of the CRL Access
22 Agent class takes the name of the CRL Database as its
23 arguments:

24 `public CRLAccessAgent(String dbName);`

25 After instantiation of a CRL Access Agent object, we can
26 call the methods of this class to obtain the information on
27 current CRL of a particular CA and check if a certificate has
28 been revoked. For example:

29 `CRLAccessAgent crlChecker;`

30 `crlChceker=new CRLAccessAgent(RevokedCert.nsf);`

1 if(!crlChecker.isRevoked(a Digital Certificate)){
2 System.out.println(The certificate is revoked!);
3 return;
4 }
5

6 Because the CRL Access Agent class use Java classes for
7 Notes Object Interface (NOI) to access the revoked certificate
8 database, the notes.jar file must be added to the class path.

9 Figure 7 shows the system for certificate revocation list
10 consolidation and access according to another preferred
11 embodiment of this invention. As shown in the Figure 7, all
12 the CRL databases can receive CRLs from CRL retriever agents,
13 and there does not exist the central database. In the
14 meantime, in order to maintain the uniform of databases, each
15 database can propagates the update to the other database. It
16 is understood that modification and variation of the
17 arrangement and the details described herein will be done by
18 others skilled in the art.

19 A method for certificate revocation list consolidation and
20 access can be obtained from the above. As shown in Figure 8,
21 in a secure network implemented by digital certificates, a
22 method for certificate revocation list (CRL) consolidation and
23 access, wherein a plurality of certificate authorities (CAs)
24 maintain and distribute the digital certificates revoked by
25 themselves in the form of CRLs, and different CAs may use
26 different CRL distribution mechanisms, said method comprising
27 the steps of: creating a plurality of CRL retrieval agents
28 based on the CRL distribution mechanisms of CAs, for
29 consolidating the CRLs from multiple CAs (801); storing the
30 consolidated CRLs from multiple CRL retrieval agents or the
31 replications of CRLs into a plurality of CRL databases (802,

1 803); and accessing the CRLs from the CRL databases by a
2 uniform set of APIs (804).

3 The present invention can be realized in hardware,
4 software, or a combination of hardware and software. The
5 present invention can be realized in a centralized fashion in
6 one computer system, or in a distributed fashion where
7 different elements are spread across several interconnected
8 computer systems. Any kind of computer system - or other
9 apparatus adapted for carrying out the methods described
10 herein - is suitable. A typical combination of hardware and
11 software could be a general purpose computer system with a
12 computer program that, when being loaded and executed,
13 controls the computer system such that it carries out the
14 methods described herein. The present invention can also be
15 embedded in a computer program product, which comprises all
16 the features enabling the implementation of the methods
17 described herein, and which - when loaded in a computer system
18 - is able to carry out these methods.

19 Computer program means or computer program in the present
20 context mean any expression, in any language, code or
21 notation, of a set of instructions intended to cause a system
22 having an information processing capability to perform a
23 particular function either directly or after conversion to
24 another language, code or notation and/or reproduction in a
25 different material form.

26 It is noted that the foregoing has outlined some of the
27 more pertinent objects and embodiments of the present
28 invention. This invention may be used for many applications.
29 Thus, although the description is made for particular
30 arrangements and methods, the intent and concept of the
31 invention is suitable and applicable to other arrangements and
32 applications. It will be clear to those skilled in the art

1 that other modifications to the disclosed embodiments can be
2 effected without departing from the spirit and scope of the
3 invention. The described embodiments ought to be construed to
4 be merely illustrative of some of the more prominent features
5 and applications of the invention. Other beneficial results
6 can be realized by applying the disclosed invention in a
7 different manner or modifying the invention in ways known to
8 those familiar with the art.

9 Thus, for example, the CRL access mechanism described in
10 the above, is independent from the CA's distribution method.
11 Although we use Lotus Domino in the preferred embodiment, it
12 is merely to illustrate the principles of the present
13 invention. It is understood that modifications employing the
14 concepts of the present invention will be apparent to others
15 skilled in the art. Therefore, the invention is to be limited
16 only by the claims.